FMS Entity Set for Work Management
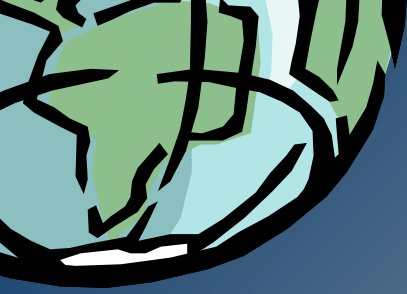
February 25, 2004

Utilities Modeling

*Compare UML to RDBMS
*Relationship to Software Engineering
*Use UML to create GDB
*Review of Previous Work
*Review of Existing Utilities Model
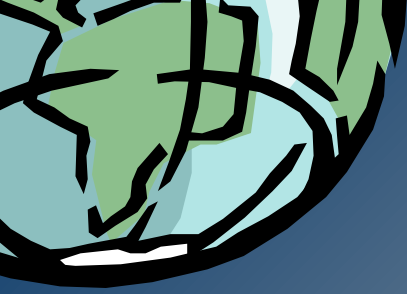*New Project Review

Relational Databases

- Optimize Database performance
- Era of Scarce Resources
  - RAM
  - Disk Space
- Normalization
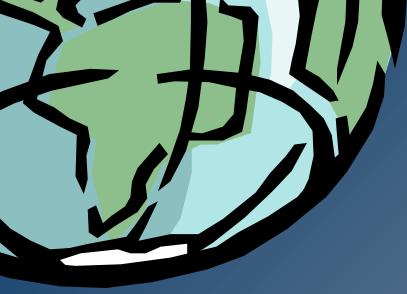 - Eliminate repeating rows

Software Engineering Crisis

- Normalization increases application complexity

- Same procedure implemented inconsistently

- Management of large applications

Object Modeling

- Create model of real world object
  - Behavior
  - Properties
- Model eliminates need for redundant procedures
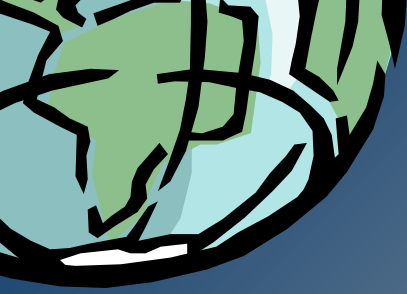- Reduces code

Object Models

- Inheritance
  - Super Class/ abstract class
  - Subclass/concrete class
  - Allows properties and behaviors to be defined once
  - Enforces standardization on children (subclasses)
  - Children carry properties of all parents
- Encapsulation
  - Insulates object from changes in other objects

UML Compared to GDB

- Both are ways to represent Object Models
- UML is Software Independent
  - can be imported into different software models
- A geodatabase (GDB)
  - is a physical implementation of an Object Model
  - May be created by UML
  - for a specific software (ESRI)
- *Most GIS SDSFIE are ESRI users*

Advantages and Disadvantages

UML or GDB?

Disadvantages

- CADD/GIS Center cannot be software specific
- Using a GDB does not necessarily promote interoperability among software applications

Advantages

- GDB can model the real world with networks
- GDB can include behaviors (topology rules)

Software Engineering Relationship

- CASE (Computer-Aided Software Engineering) Tools
  - The use of computer-based support in the software development process
- To create UML, one can use a variety of applications including Visio Professional

Create a GDB with UML

- Schema Creation Wizard

  – Connects to a Repository (mdb) or XMI (export file) generated from Visio

- The schema of the GDB (including domains, relationships, connectivity, etc.) is implemented through ArcCatalog

- Requires ArcEditor or ArcInfo license
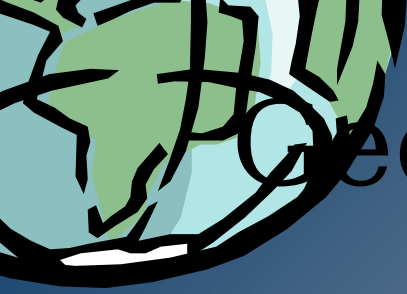
Geodatabase Components

- Associations exist in the form of relationships
  - Relationship class
  - Linked annotation
- Non-graphic information is stored in separate tables
  - Layer file

ESRI Views

- A Layer File creates a "view" of the data
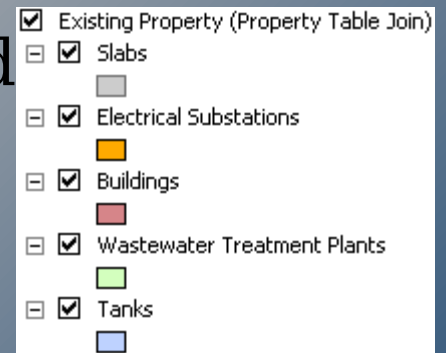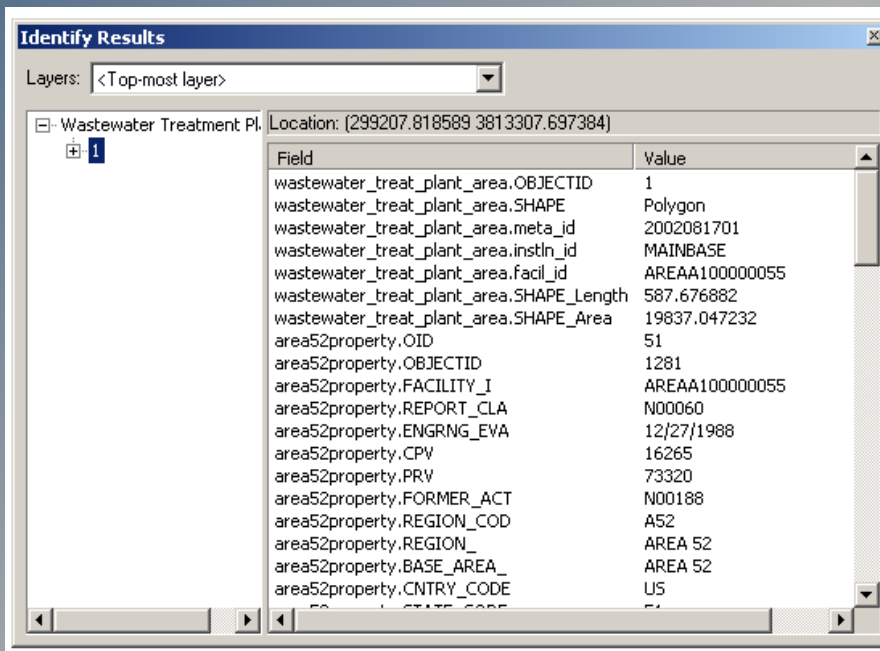 - A user's perspective can be generated regardless of the underlying data structure
  - Complexity is hidden from the end

Previous Work Entity Set

- Facilities
  - Simple
  - Linkages to standalone tables
- Utilities
  - Scale
  - Aggregation

Asset

#FacilityNumber : String

Work

+WorkOrderID : String
+UniqueID : Double
+ProjectTitle : String
+Status : String
+StatusDate : Date
+Description : String
+FacilityNumber : String
+Location : String
+ReasonWO : String
+ChangeNum : String
+ChangeTitle : String
+ChangeBy : String
+ChangeDate : String
+ReasonCO : String
+Parent : String
+HasChildren : Boolean
+Contract : String
+WOPriority : String
+WOSequence : String
+FollowUp : Boolean
+FollowUpfromWO : String
+WorkType : String
+AssetObjID : String
+TeamOrgID : String
+CustOrgID : String
+Emergency : Boolean
+OversightOrg : String
+JustificationNote : String
+CreatedByPerson : String
+CustomerPOC : String
+BudgetAmt : Currency
+BudgetAcctNumber : String
+CreateDate : Date
+FiscalYearRequested : short
+FiscalYearComplete : short
+EstimatedCost : Currency
+BeginDate : Date
+EndDate : Date
+OwnershipCode : String
+RespOfficer : String
+ActualCost : Currency
+Restrictions : String
+BaseCode : String
+BaseName : String

Facility Work Orders

- Strategy
  - Compare existing models
  - Compile to identify commonalities and unique attributes of each model into a series of Excel spreadsheets
  - Generate diagrams of proposed model (a comprehensive schema of all models)
  - Ensure linkages (to other packages and SDS)

Facility Work Orders

Sample of the Draft Work Execution Framework

Construction

+LeadCraft : String
+EstDuration : Double
+EstLabHours : Double
+EstMatCost : Double
+EstLabCost : Double
+EstToolCost : Double
+ActLabHours : Double
+ActToolCost : Double
+OutLabCost : Double
+OutMatCost : Double
+OutToolCost : Double
+TargetCompDate : Date
+TargetStartDate : Date
+InterruptTab : String
+Downtime : String
+CrewID : String
+Supervisor : String
+ConstComplRate : short
+ConstMethod : String
+DesignInstuctComments : String
+DesignInstructionDate : Date
+DesignInstructionNum : Boolean
+ArchHistPermit : Boolean
+AirQualityPermit : Boolean
+EnvironConstPermit : Boolean
+FAACert : Boolean
+FAADate : Date
+SitingApproval : Boolean
+SitingDate : Date
+WetlandCert : Boolean
+TypeConstCode : String
+ConstRemarks : String

New

Alteration

+ProblemCode : String
+InstRestorationPermit : Boolean
+RehabCost : short
+ReplaceCost : short
+Cond_D : int

Improvement

-ProblemCode : String
-InstRestorationPermit : Boolean
-RehabCost : short
-ReplaceCost : short
-Cond_D : int

Yokosuka Utilities Model

- To promote interoperability among Naval commands
  - Pacific Spatial Data Consortium
- Early development stage of Water, Electric, and Gas utility models with basic UML class diagrams
- Evaluate Water, Gas, Electric, Storm Water, HCS, and Fuel utilities
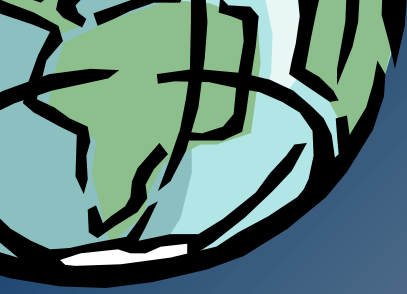
Basic Data Model Structure

Utility Model

Utility Resource

Component | Facility | Support | Management

Component

- An object that has a direct function of generation, control, or distribution of the utility and is essential to the utility network.

- Example:

     Pipes, valves, pumps, fittings

Facility

- An aggregation of components which may itself serve as a component of the utility network.
- Example:

  Pump station, treatment plant
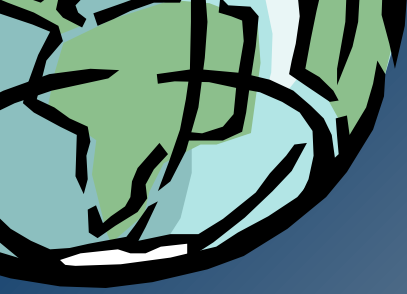
Support

- An object that enables or protects components of the utility network.

- Example:

    Casing, thrust protector, anode

Management (Facilities)

- Facilities (physical) – Objects that support the operation of utilities at an installation like a military base, large office complex, or municipality. These objects may be spatially referenced with a physical location.
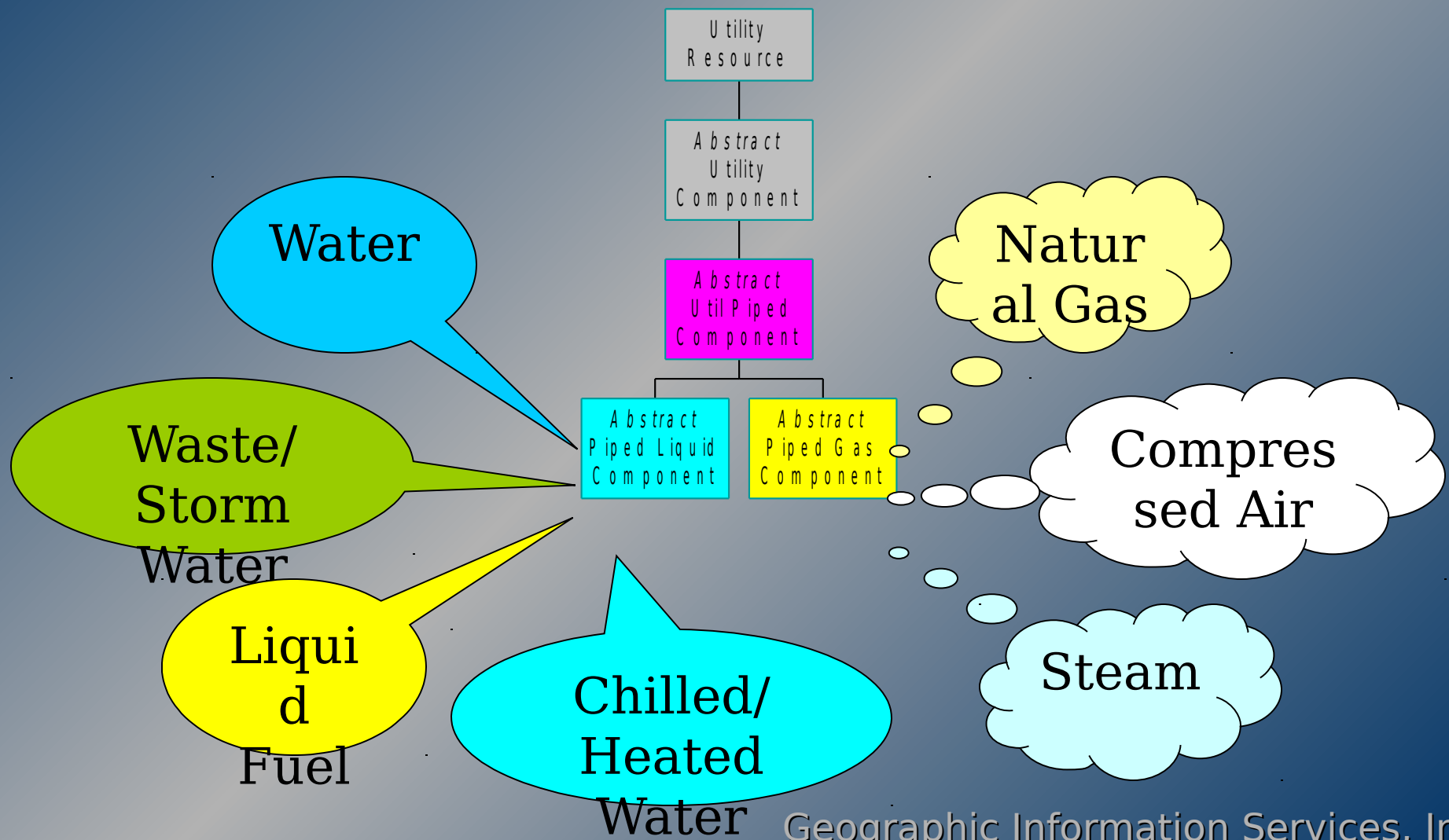
- Example:
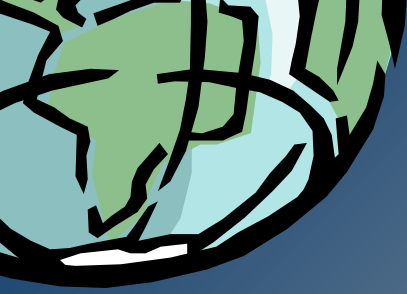
  Leak locations, repair locations

Management (Utilities)

- Utilities (logical) – Objects that support the utilities business processes and are not spatially referenced.

- Example:

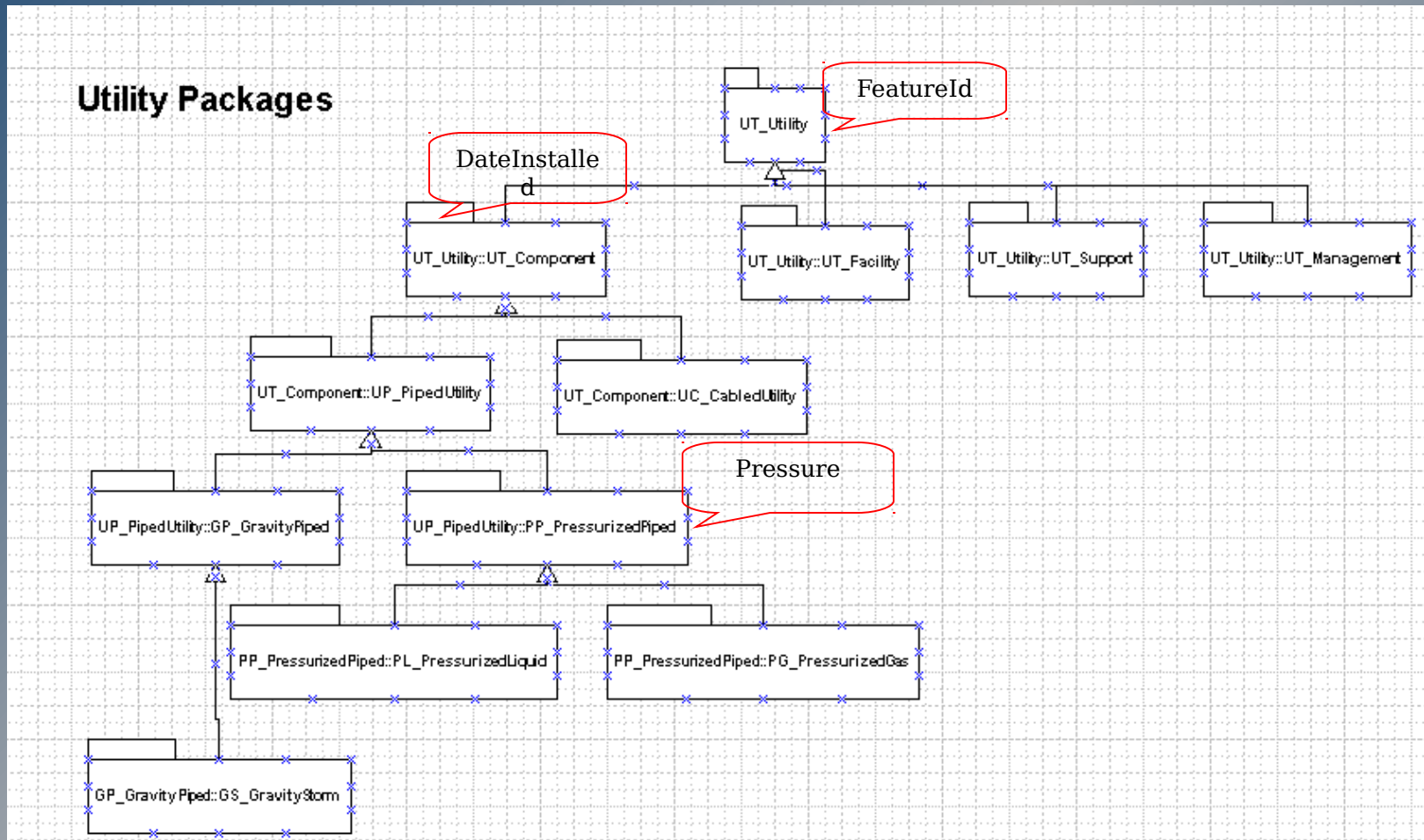   Customer service data, circuit data

Common Utility Objects

Utility Resource

Abstract Utility Component

Abstract Util Piped Component

Abstract Piped Liquid Component

Abstract Piped Gas Component

Water

Waste/ Storm Water

Liquid Fuel

Chilled/ Heated Water

Natural Gas

Compressed Air

Steam

Utilities Diagram

New Project Review

- Structure
- Classes (Feature classes)
- Attributes

Methods

- Compare with existing SDSFIE
- Identify high level objects
- Identify Inconsistencies across utilities
- Realign item names
- Standardize domains